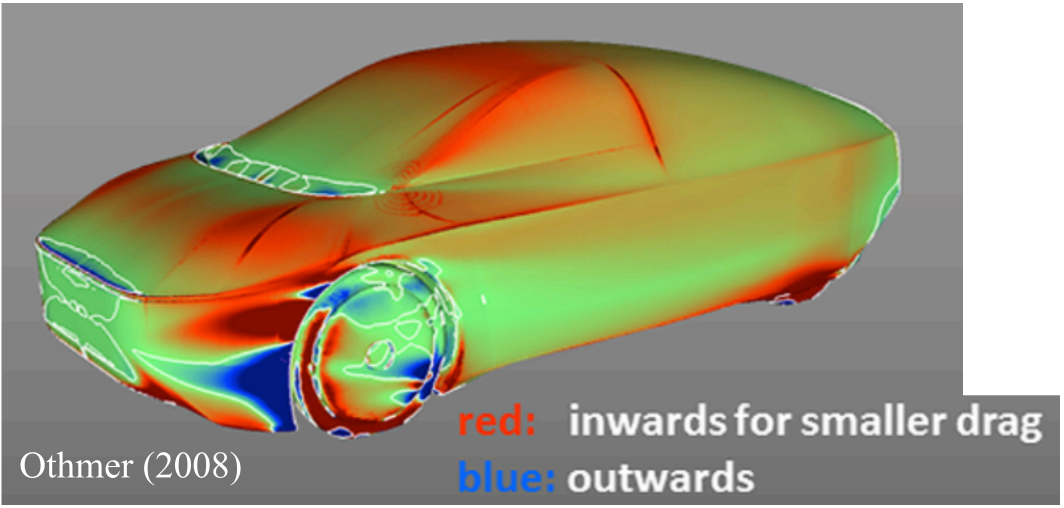
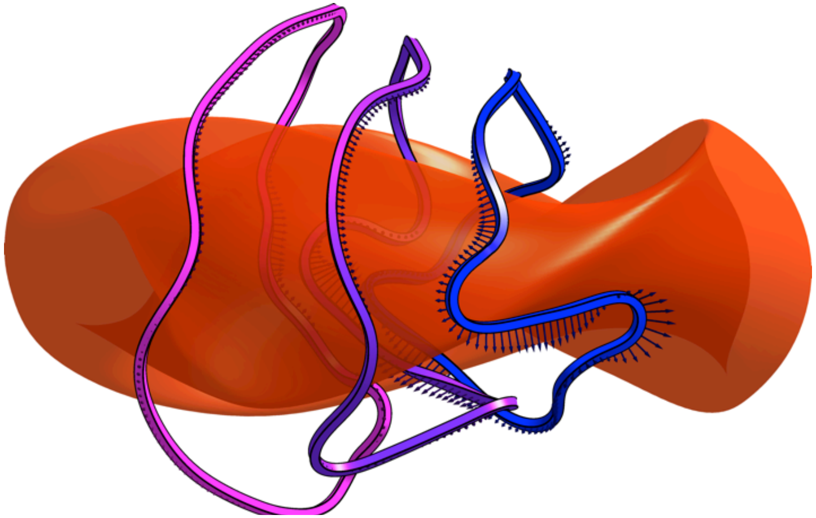
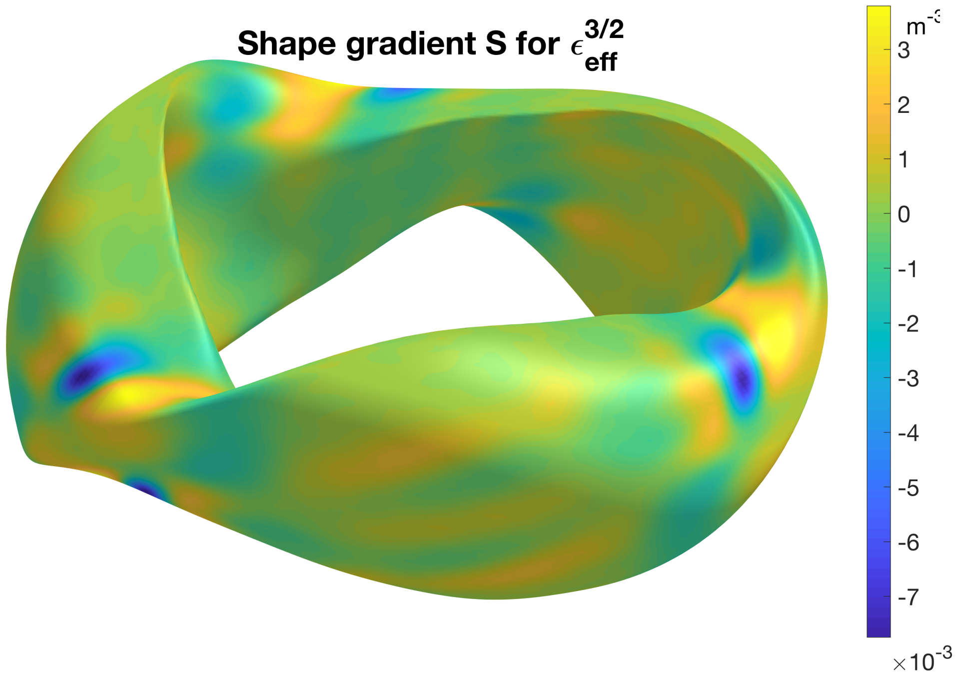
Understanding local sensitivity & tolerances of stellarators using shape gradients

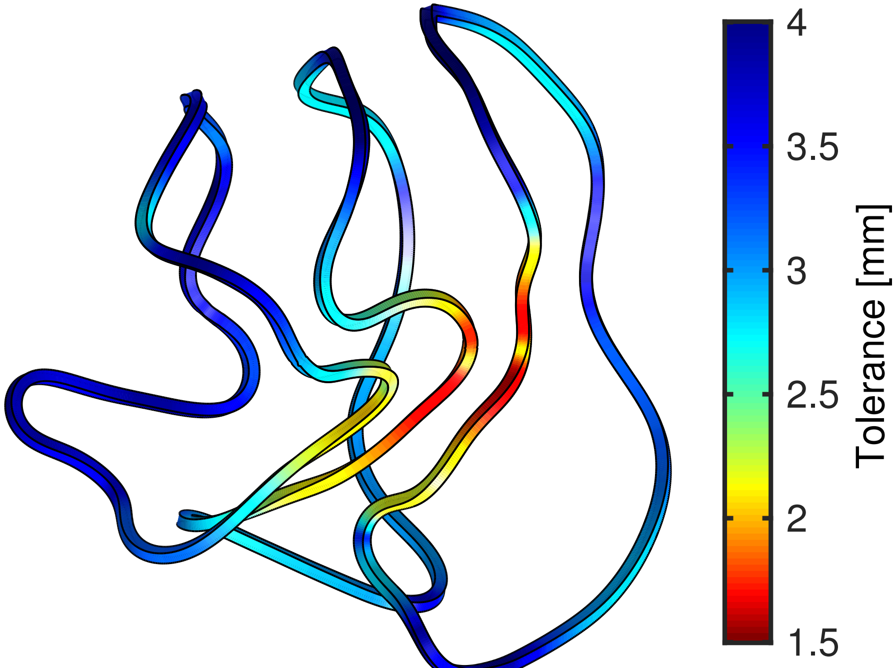
Matt Landreman, Elizabeth Paul, U of Maryland

Nuclear Fusion 58, 076023 (2018)









The shape gradient is a new (to us) way to think about derivatives involving shapes.

- Derivatives involving shapes are central to stellarator optimization.
- These derivatives also encode tolerances, which have been a leading driver of cost:

  "The largest driver of the project cost growth were the accuracy requirements." [Strykowsky et al, *Engineering Cost & Schedule Lessons Learned on NCSX,* (2009)]

- Compared to 'parameter derivatives', shape gradients have 2 advantages:
  - *Spatially local*
  - *Independent of parameterization*

Understanding local sensitivity of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from parameter derivatives (e.g. STELLOPT)
- Examples
  - Rotational transform
  - Neoclassical transport
  - Coil-plasma distance
- Coil tolerances
- Magnetic sensitivity and tolerances

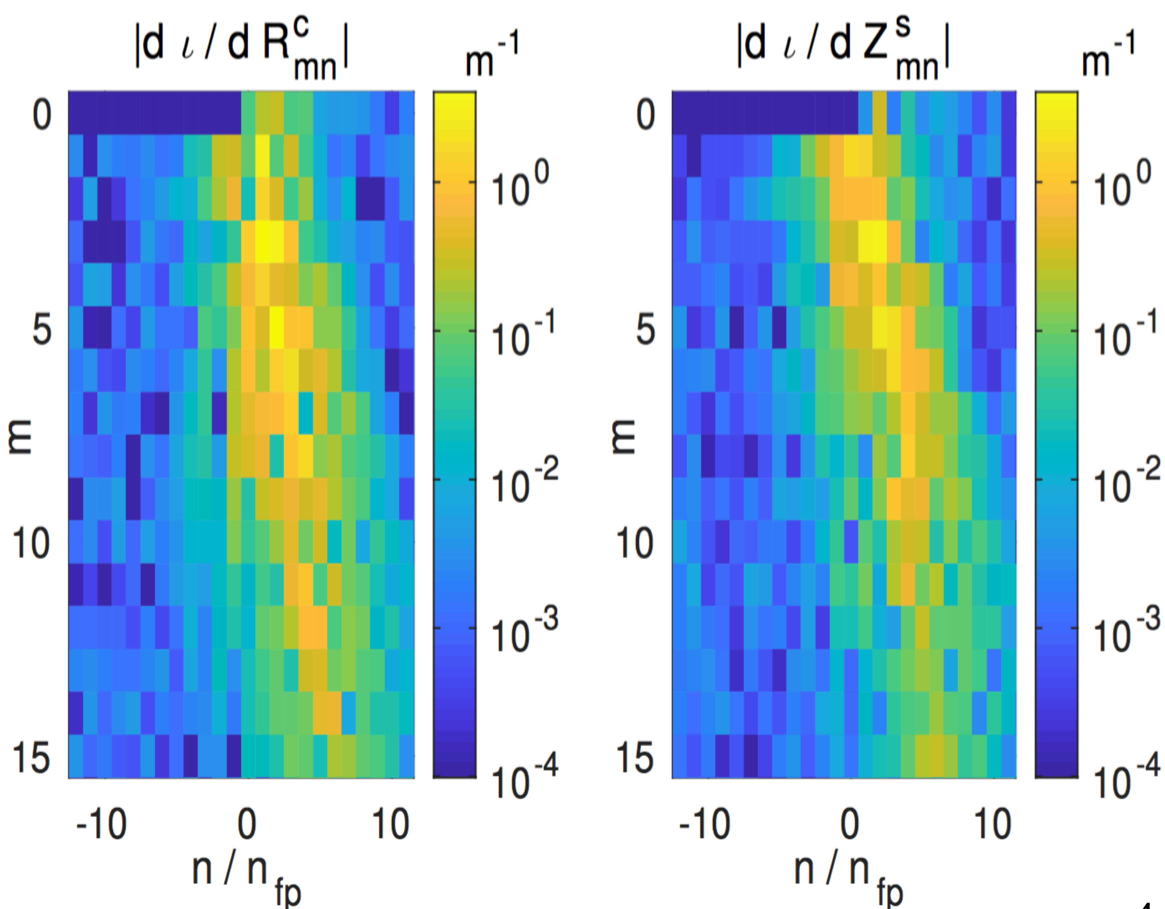
Parameter derivatives and shape gradients are complementary ways to express derivatives.

Let f denote any figure of merit, e.g. ι, neoclassical transport, etc.

Parameter derivatives: Example: $\partial f / \partial R^c_{m,n}$ and $\partial f / \partial Z^s_{m,n}$ where $R^c_{m,n}$ and $Z^s_{m,n}$ parameterize the plasma boundary shape:

$$R(\theta,\zeta) = \sum_{m,n} R^c_{m,n} \cos(m\theta - n\zeta),$$

$$Z(\theta,\zeta) = \sum_{m,n} Z^s_{m,n} \sin(m\theta - n\zeta)$$



Parameter derivatives and shape gradients are complementary ways to express derivatives.

Let f denote any figure of merit, e.g. ι, neoclassical transport, etc.

Parameter derivatives: Example: $\partial f / \partial R^c_{m,n}$ and $\partial f / \partial Z^s_{m,n}$

where $R^c_{m,n}$ and $Z^s_{m,n}$ parameterize the plasma boundary shape:

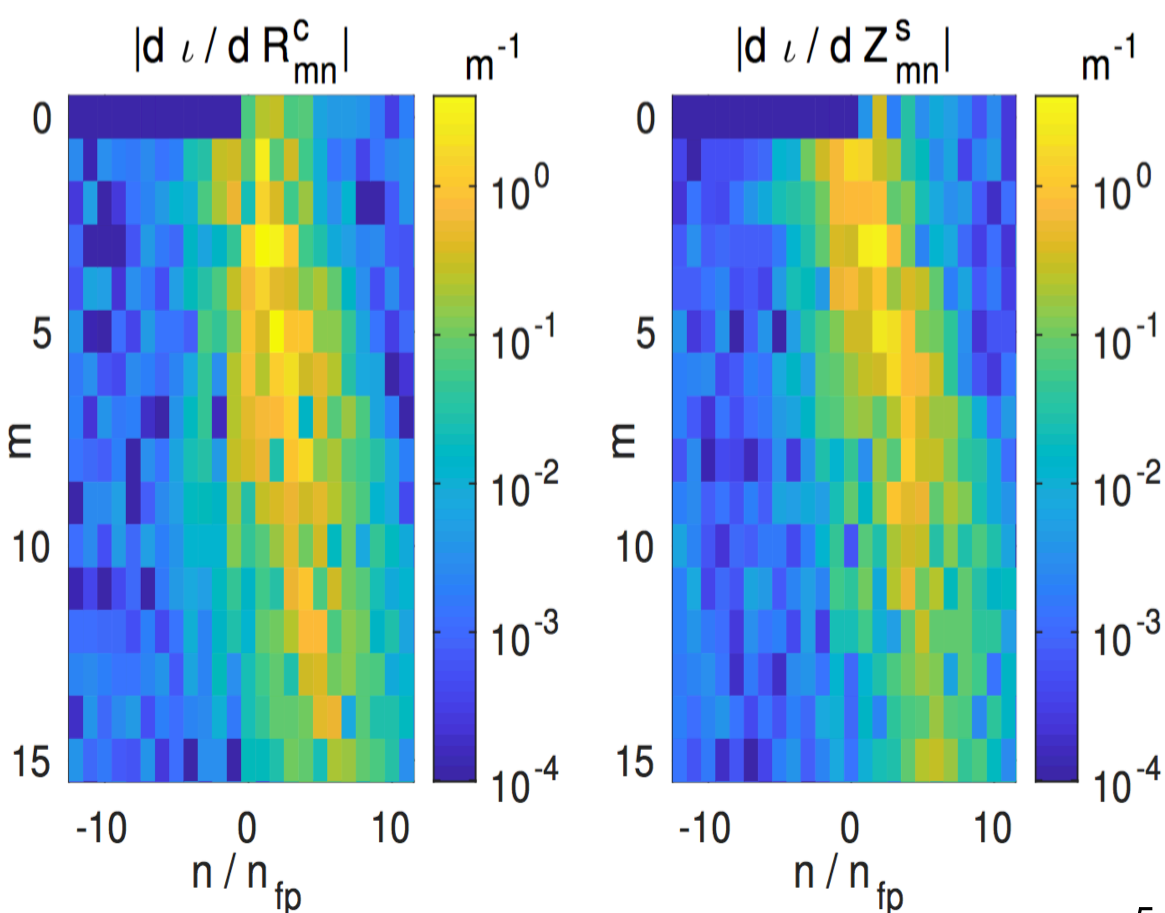
$$R(\theta,\zeta) = \sum_{m,n} R^c_{m,n} \cos(m\theta - n\zeta),$$

$$Z(\theta,\zeta) = \sum_{m,n} Z^s_{m,n} \sin(m\theta - n\zeta)$$

- Successfully used in STELLOPT to design NCSX, etc.
- Computable by finite differencing any code.

But,

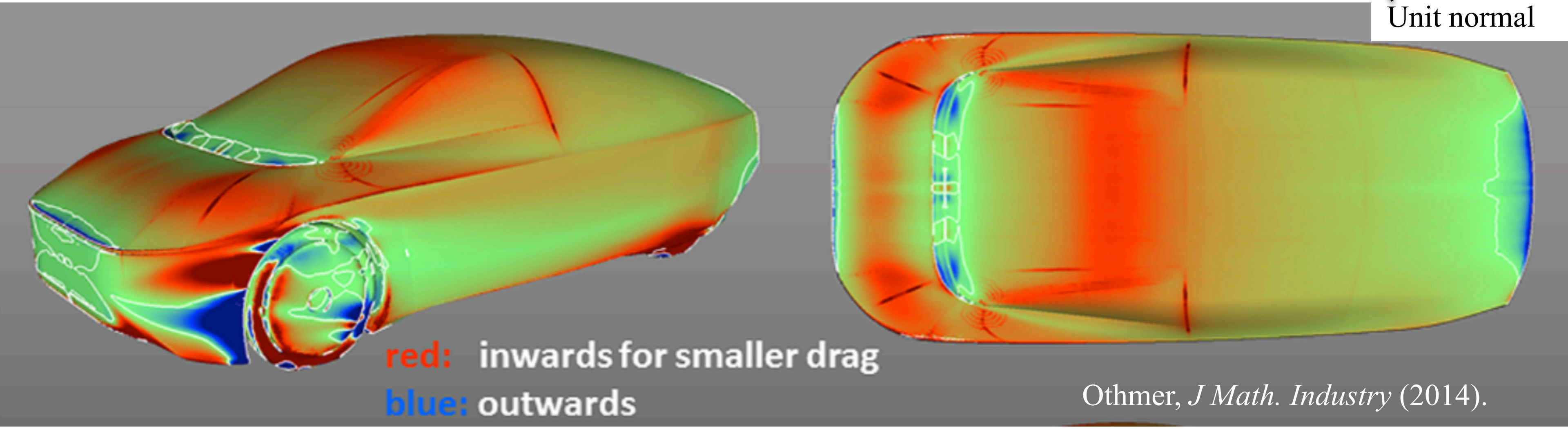
- Not unique: coordinate-dependent,
- Nonlocal: awkward for engineering.



Parameter derivatives and shape gradients are complementary ways to express derivatives.

Shape gradients: For surfaces, S where $\delta f = \int d^2a \left(\delta \mathbf{r} \cdot \mathbf{n} \right) S$

Unit normal



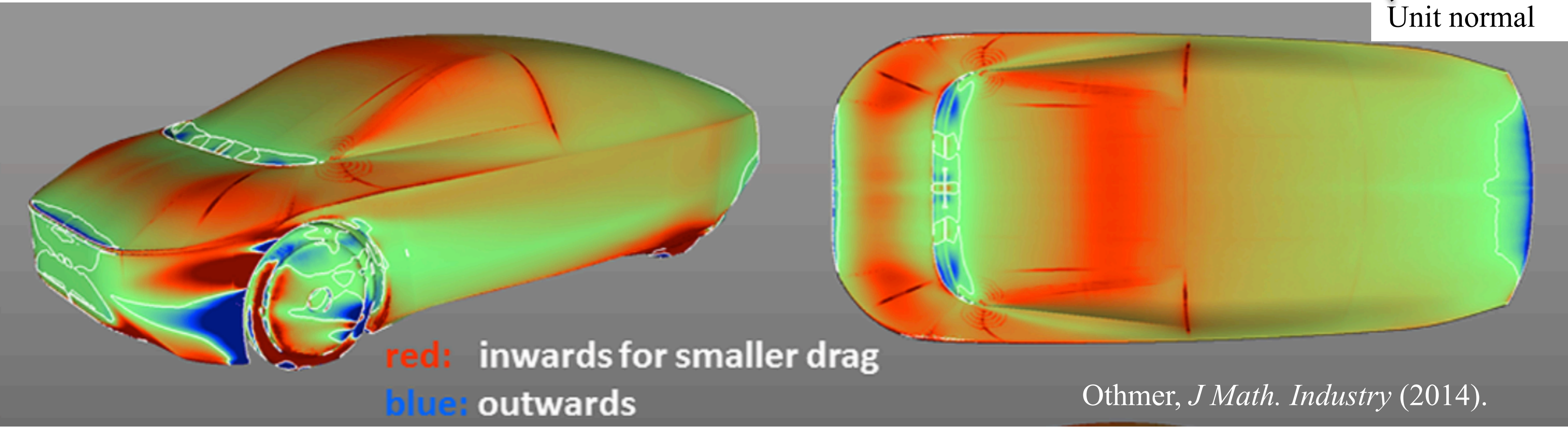
Othmer, *J Math. Industry* (2014).

- Local (real-space, not Fourier-space). More useful for engineering.
- Independent of coordinates used to parameterize surface.
- May provide different insight than parameter derivatives.

Parameter derivatives and shape gradients are complementary ways to express derivatives.

Shape gradients: For surfaces, S where $\delta f = \int d^2a \left(\delta\mathbf{r} \cdot \mathbf{n} \right) S$

Unit normal



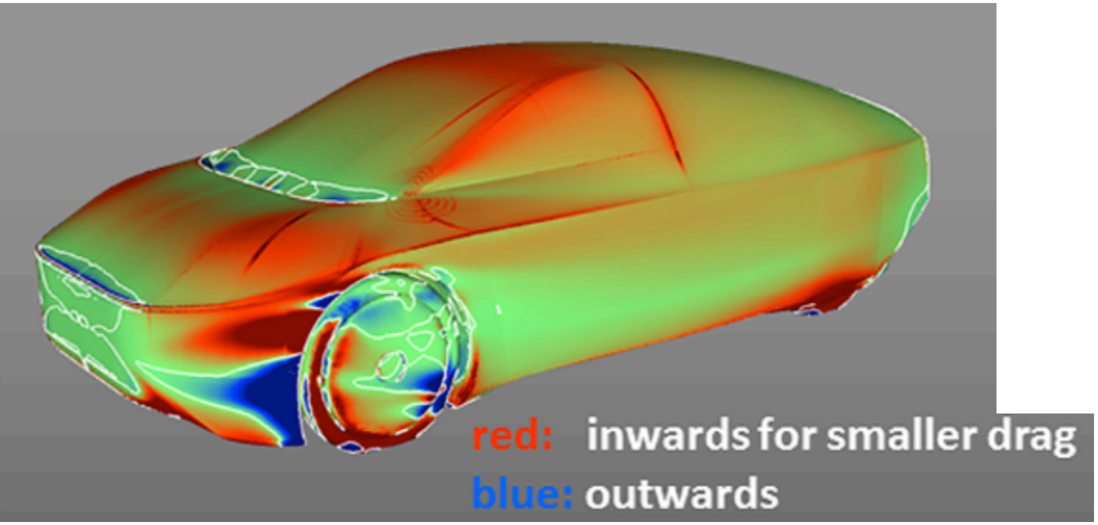
Othmer, *J Math. Industry* (2014).

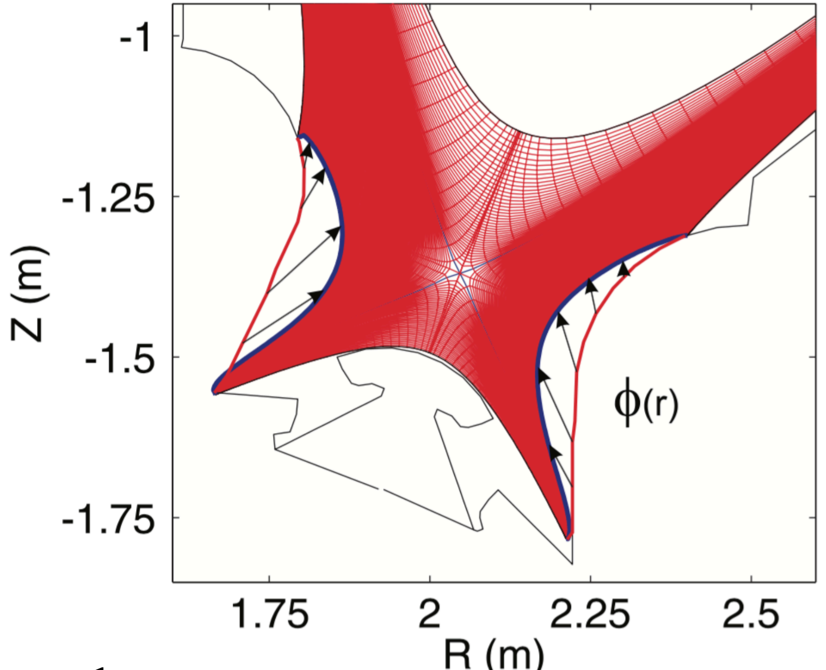
- Local (real-space, not Fourier-space). More useful for engineering.
- Independent of coordinates used to parameterize surface.
- May provide different insight than parameter derivatives.

For coils, $\mathbf{S}_k$ where $\delta f = \sum_{\text{coils } k} \int d\ell \; \delta\mathbf{r} \cdot \mathbf{S}_k$, and $\mathbf{S}_k \cdot \frac{\partial \mathbf{r}}{\partial \ell} = 0$.

Parameter derivatives and shape gradients are complementary ways to express derivatives.

Shape gradients: For surfaces, S where $\delta f = \int d^2a \left(\delta \mathbf{r} \cdot \mathbf{n} \right) S$





Recently used for optimizing tokamak divertor shapes:

- W. Dekeyser, Ph.D. thesis, KU Leuven (2014).
- W. Dekeyser et al, *Nucl. Fusion* 54, 073022 (2014).
- M. Baelmans, et al, *Nucl. Fusion* 57, 036022 (2017).

The shape gradient representation can be expected to exist for many important shape functionals.

1D:

Derivative of a function of n numbers $f\left(r_1, r_2, \ldots, r_n\right)$:

$$\delta f = \sum_{j=1}^{n} \frac{\partial f}{\partial r_j} \delta r_j$$

$$n \to \infty \text{ limit: } \; f = f\left[r\left(\ell\right)\right], \quad \delta f = \int d\ell \underbrace{\frac{\delta f}{\delta r}}_{S} \delta r$$

This is an instance of the Riesz representation theorem:
Any linear operator can be written as an inner product with some element of the appropriate space.

In some cases, shape gradients can be computed analytically.

Integrals over a curve:

If $f\left[C\right]=\int_C d\ell\, Q$ for some $Q\left(\mathbf{r}\right)$ and space curve C,

$$\Rightarrow \quad \delta f=\int_C d\ell\, \delta\mathbf{r}\cdot\underbrace{\left[\left(\overleftrightarrow{\mathbf{I}}-\mathbf{tt}\right)\cdot\nabla Q-Q\kappa\mathbf{n}\right]}_{\mathbf{s}}$$

where $\kappa=$ curvature, $\mathbf{t}=$ tangent.

Integrals over a surface:

If $f\left[\partial\Omega\right]=\int_{\partial\Omega} d^2a\, Q$ for some $Q\left(\mathbf{r}\right)$ and surface $\partial\Omega$,

$$\Rightarrow \quad \delta f=\int_{\partial\Omega} d^2a\left(\delta\mathbf{r}\cdot\mathbf{n}\right)\underbrace{\left(\mathbf{n}\cdot\nabla Q-2QH\right)}_{S}$$

where $H=$ mean curvature.

Understanding local sensitivity of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from parameter derivatives (e.g. STELLOPT)
- Examples
  - Rotational transform
  - Neoclassical transport
  - Coil-plasma distance
- Coil tolerances
- Magnetic sensitivity and tolerances

Previously, shape gradients have usually been computed using adjoint methods.

- An adjoint equation, when available, gives the shape gradient (representing all possible shape perturbations) at the cost of ~ 1 regular solve.
- Then, the shape gradient is very efficient for shape optimization:

$$\text{Choose } \delta\mathbf{r} = -\varepsilon S\mathbf{n} \quad \Rightarrow \quad \delta f = -\varepsilon \int d^2a \, S^2 < 0.$$

Step size $\varepsilon > 0$

- However, adjoint methods require analytic work & code development for every target.
- Here, instead, we show how to compute shape gradients from any code with little extra work.

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \qquad \& \; Y, Z$$

Parameters p_j are $\left\{ X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s \right\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \quad \& \; Y, Z$$

Parameters p_j are $\{X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

Discretize shape gradient:

$$S_X(\vartheta) = S_{X,0}^c + \sum_{m=1} \left[S_{X,m}^c \cos(m\vartheta) + S_{X,m}^s \sin(m\vartheta) \right] \quad \& \; S_Y, S_Z$$

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \qquad \& \; Y, Z$$

Parameters p_j are $\{X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

Discretize shape gradient:

$$S_X(\vartheta) = S_{X,0}^c + \sum_{m=1} \left[S_{X,m}^c \cos(m\vartheta) + S_{X,m}^s \sin(m\vartheta) \right] \qquad \& \; S_Y, S_Z$$

$$\int d\ell \, \delta\mathbf{r} \cdot \mathbf{S} = \delta f \qquad \Rightarrow \qquad \text{Solve } \int d\ell \, \frac{\partial \mathbf{r}}{\partial p_j} \cdot \mathbf{S} = \frac{\partial f}{\partial p_j} \text{ for } \mathbf{S}.$$

(Square linear system)

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Coils: Discretize coil shapes:

$$X(\vartheta) = X_0^c + \sum_{m=1} \left[X_m^c \cos(m\vartheta) + X_m^s \sin(m\vartheta) \right] \qquad \& \; Y, Z$$

Parameters p_j are $\{X_m^c, X_m^s, Y_m^c, Y_m^s, Z_m^c, Z_m^s\}$.

Compute $\partial f / \partial p_j$ using finite differences, e.g. STELLOPT.

Discretize shape gradient:

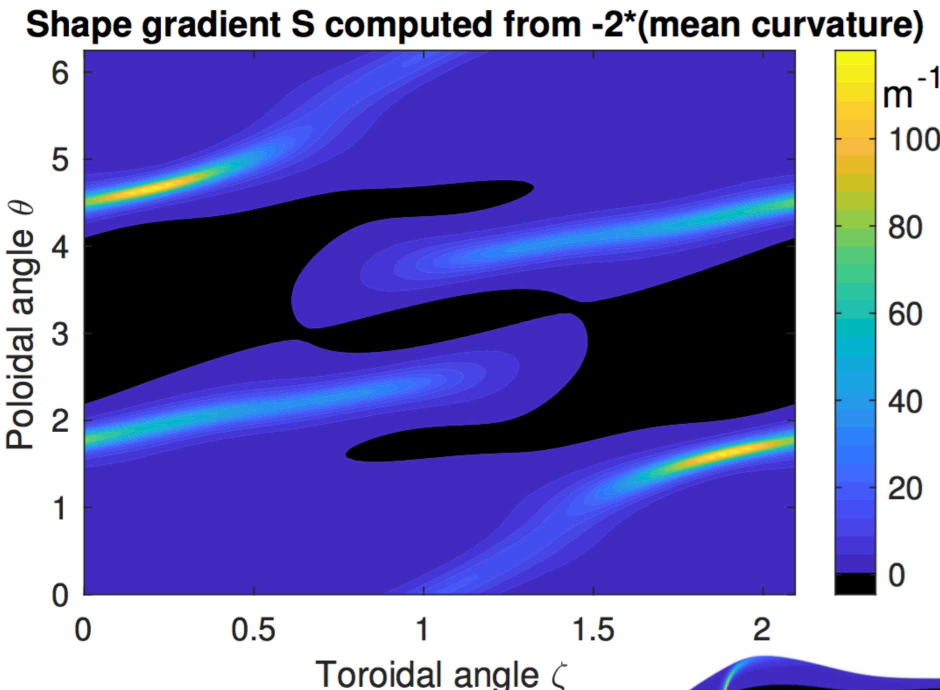
$$S_X(\vartheta) = S_{X,0}^c + \sum_{m=1} \left[S_{X,m}^c \cos(m\vartheta) + S_{X,m}^s \sin(m\vartheta) \right] \qquad \& \; S_Y, S_Z$$

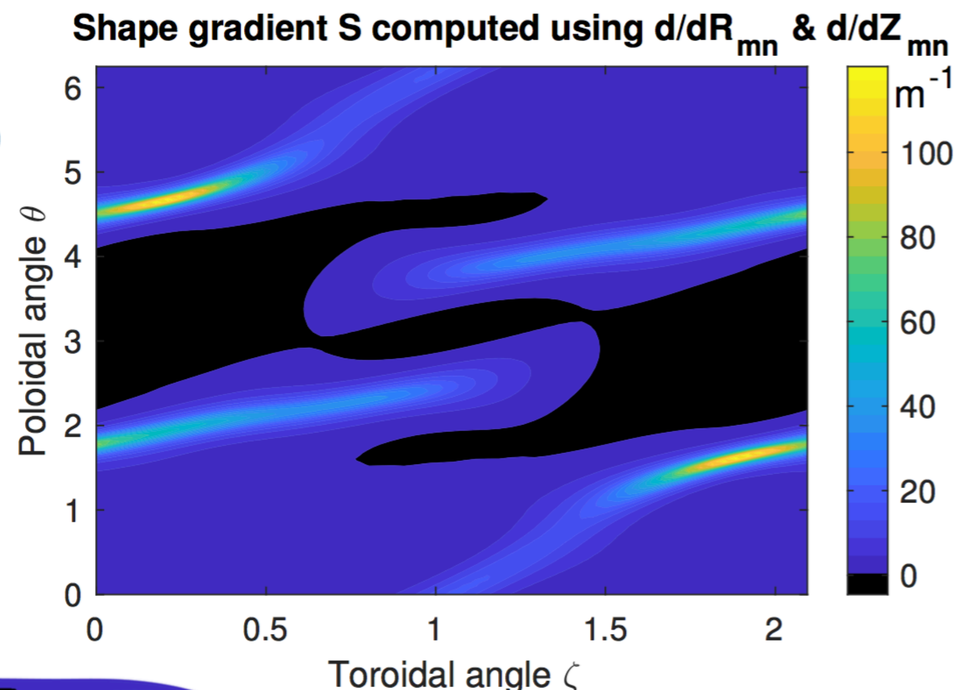
$$\int d\ell \, \delta\mathbf{r} \cdot \mathbf{S} = \delta f \qquad \Rightarrow \qquad \text{Solve } \int d\ell \, \frac{\partial \mathbf{r}}{\partial p_j} \cdot \mathbf{S} = \frac{\partial f}{\partial p_j} \text{ for } \mathbf{S}.$$

Similar procedure for boundary surface.

The algorithm for computing shape gradients can be verified by comparison to analytic theory.

Consider $f = \text{area}$. Analytic result: $S = -2 \times \left(\text{mean curvature}\right)$





$$\delta f = \int d^2 a \left(\delta \mathbf{r} \cdot \mathbf{n}\right) S$$

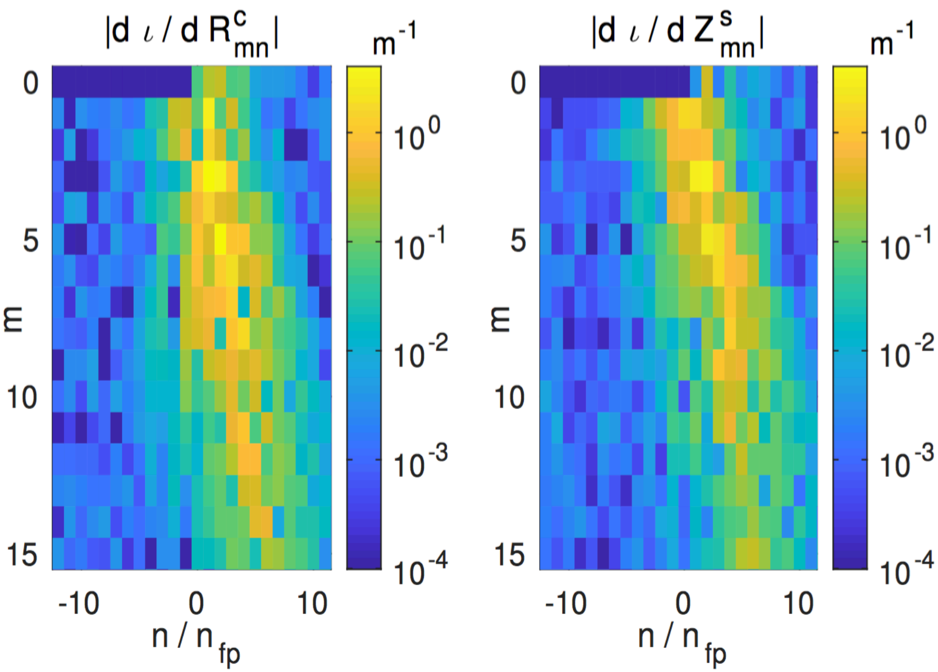
Understanding local sensitivity of stellarators using shape gradients

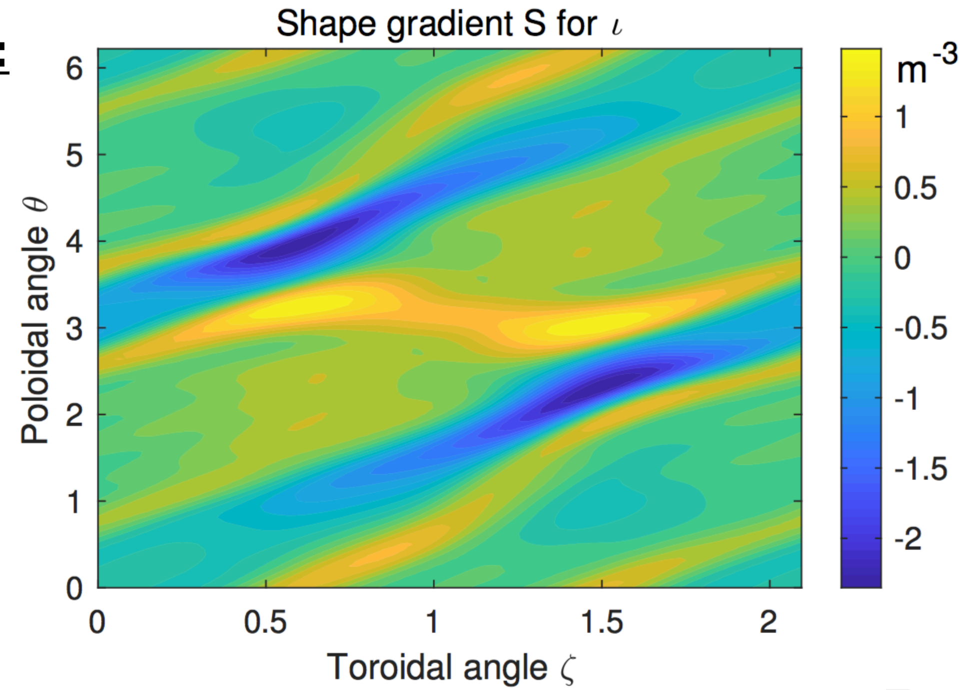
- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from parameter derivatives (e.g. STELLOPT)
- Examples
  - Rotational transform
  - Neoclassical transport
  - Coil-plasma distance
- Coil tolerances
- Magnetic sensitivity and tolerances

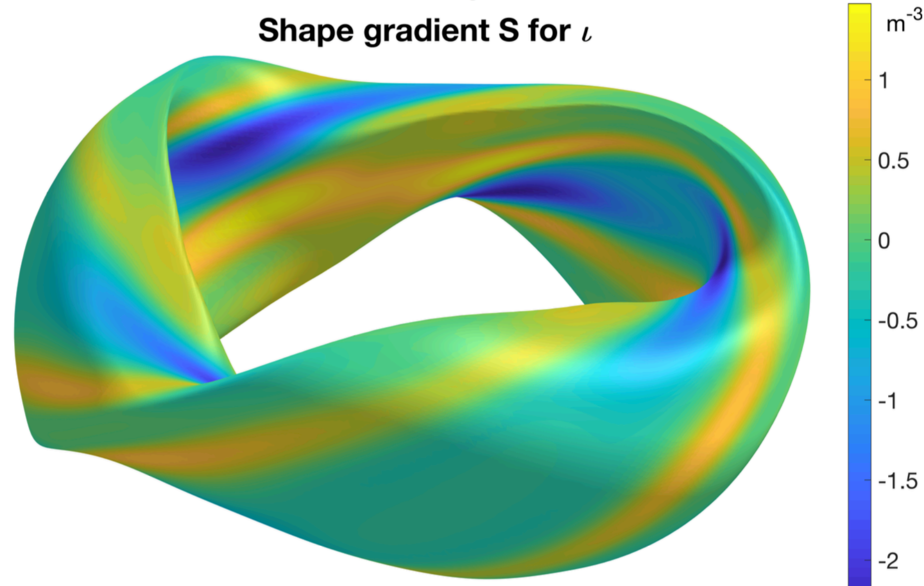
Example: Rotational transform at r/a=0.5

<u>Shape gradient for boundary surface:</u>

Parameter derivatives from STELLOPT/VMEC:



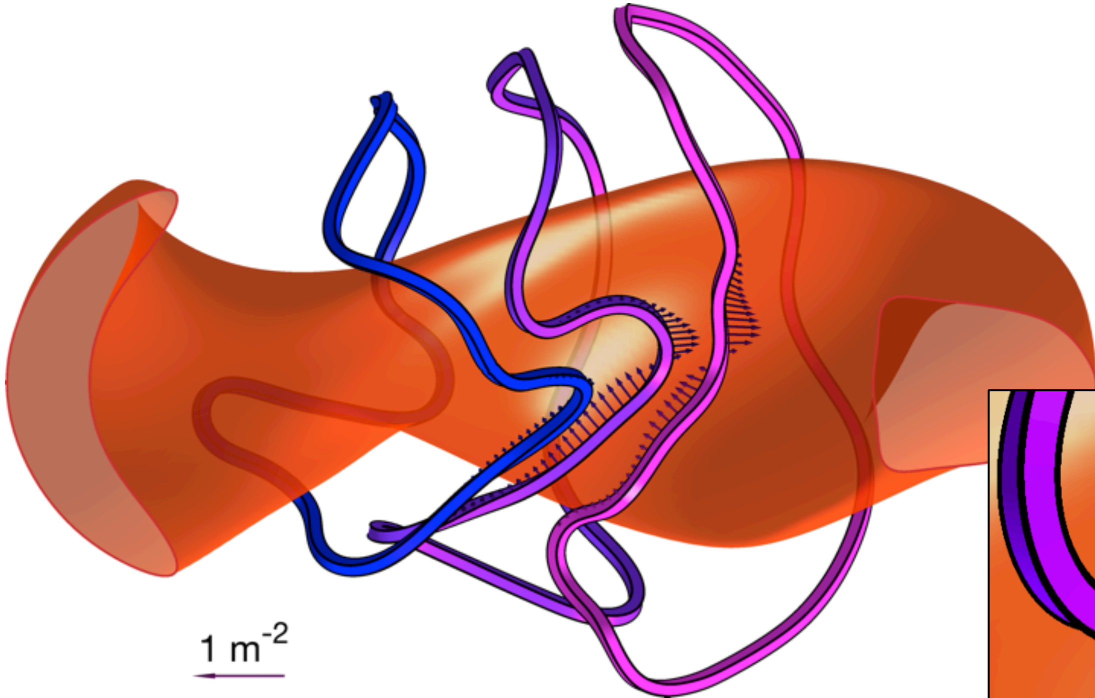




$$\delta f = \int d^2 a \left(\delta \mathbf{r} \cdot \mathbf{n} \right) S$$

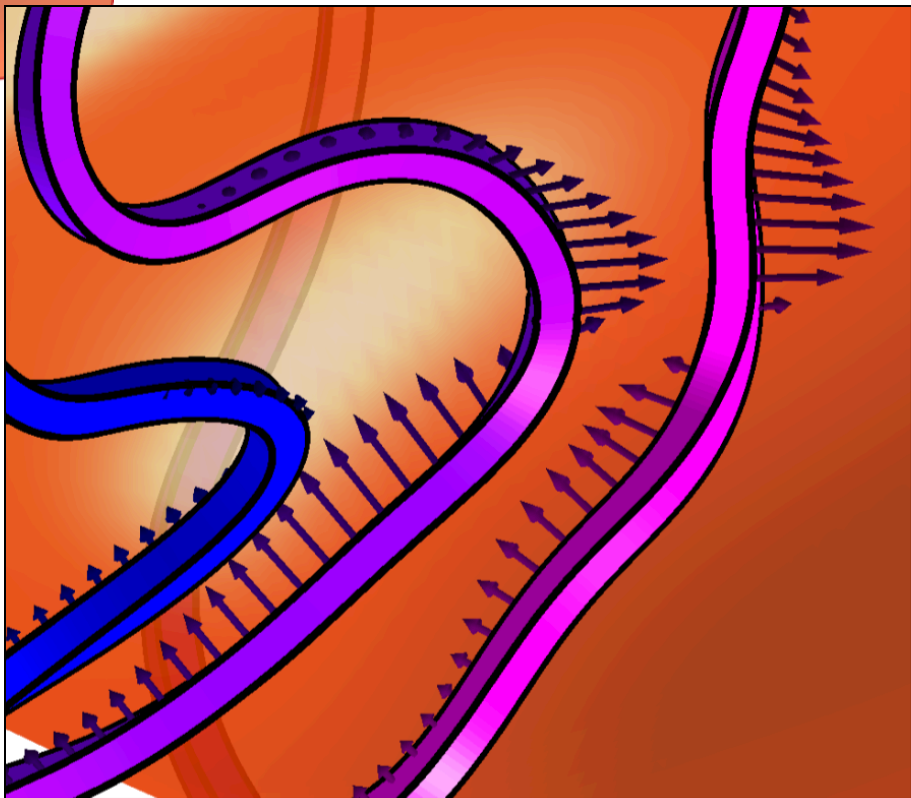
Example: Rotational transform at r/a=0.5

Shape gradient for coils:



Arrows show $\mathbf{S}_k$ where

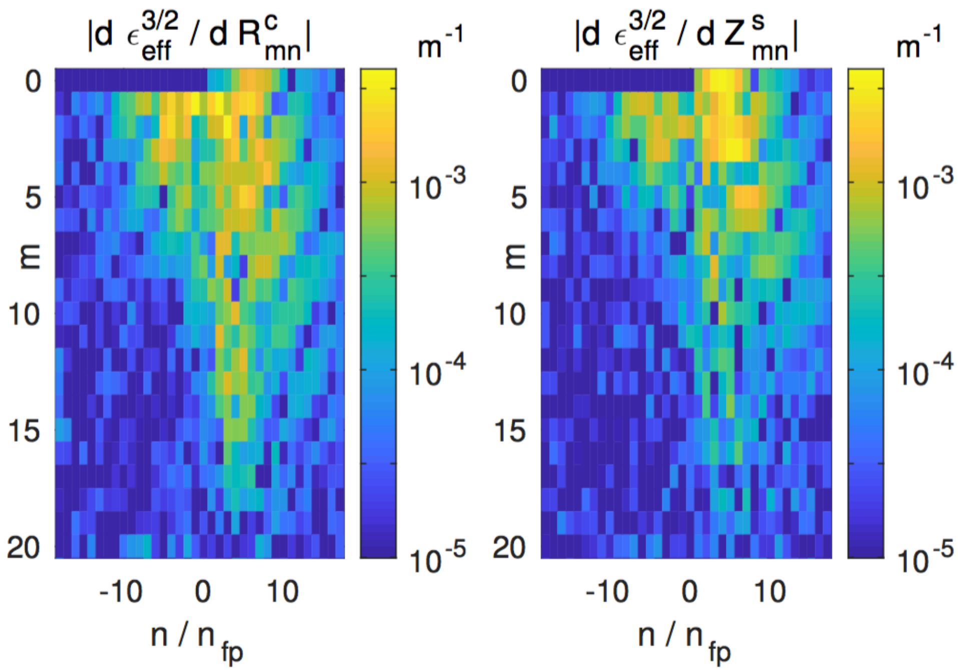
$$\delta f = \sum_{\text{coils } k} \int d\ell \; \delta\mathbf{r}\cdot\mathbf{S}_k .$$



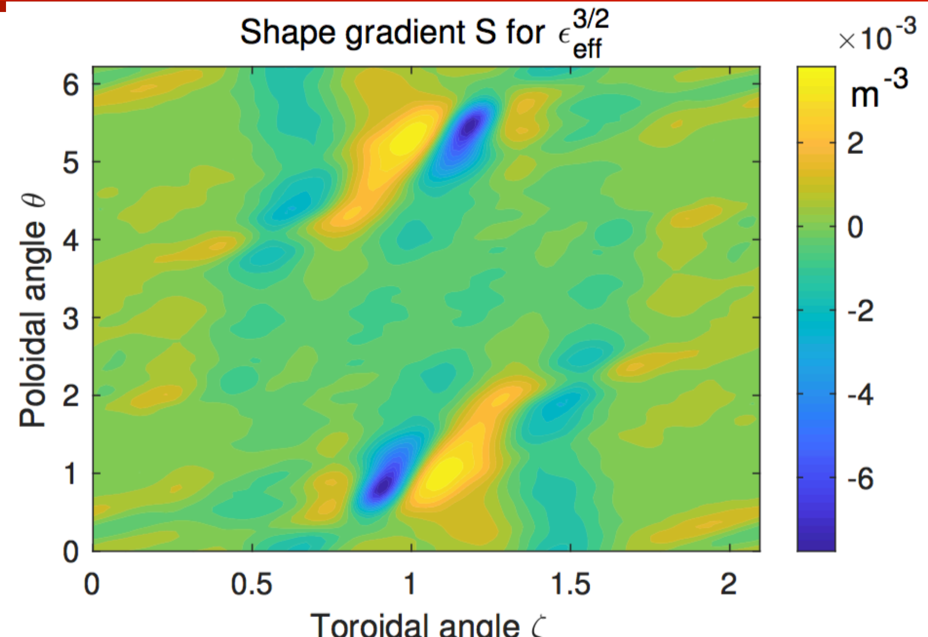
Example: Neoclassical transport $\varepsilon_{eff}^{3/2}$ at r/a=0.5

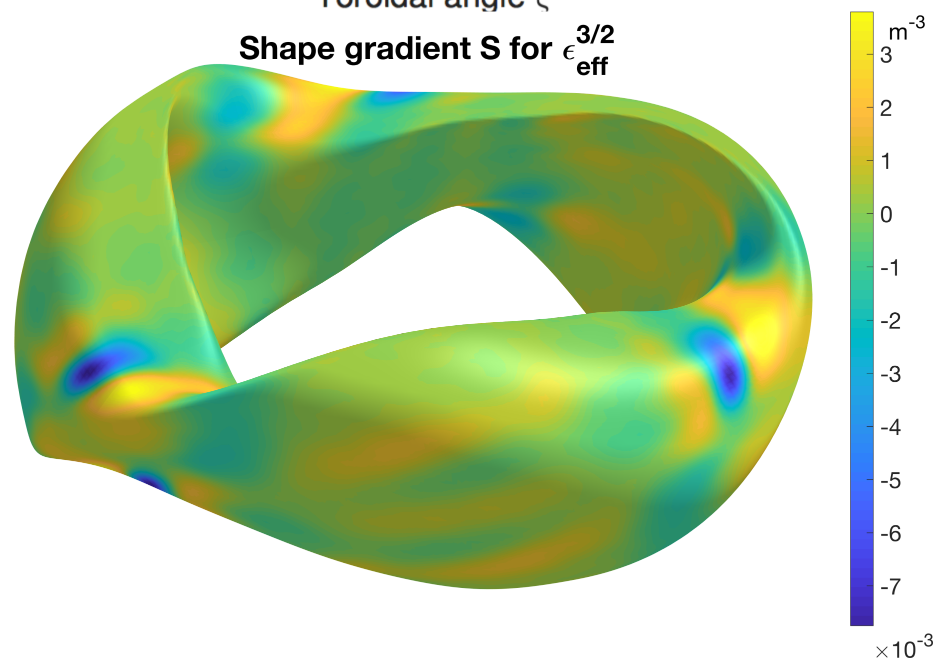
Shape gradient for boundary surface:

Parameter derivatives from STELLOPT/VMEC/NEO:



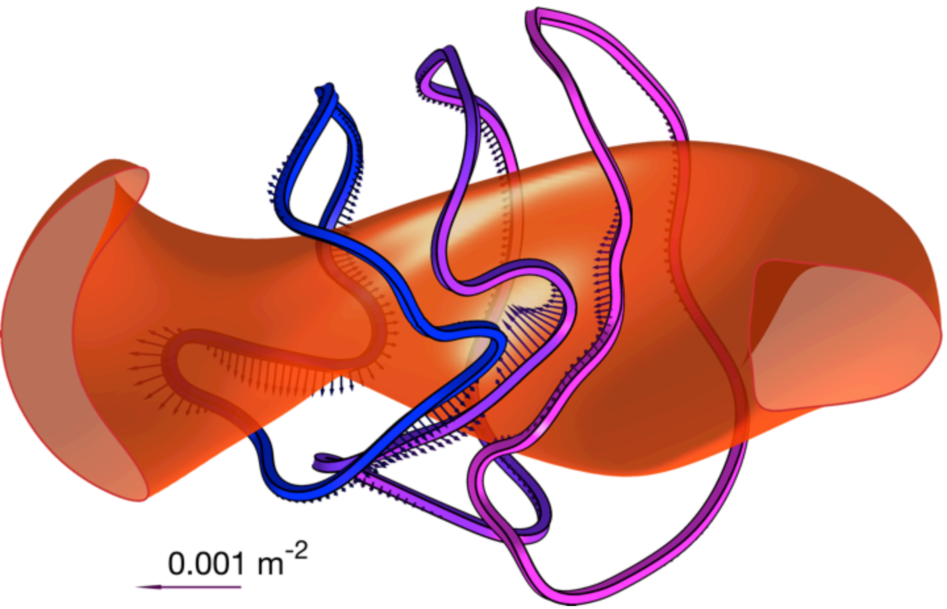
$$\delta f = \int d^2a \left(\delta \mathbf{r} \cdot \mathbf{n}\right) S$$

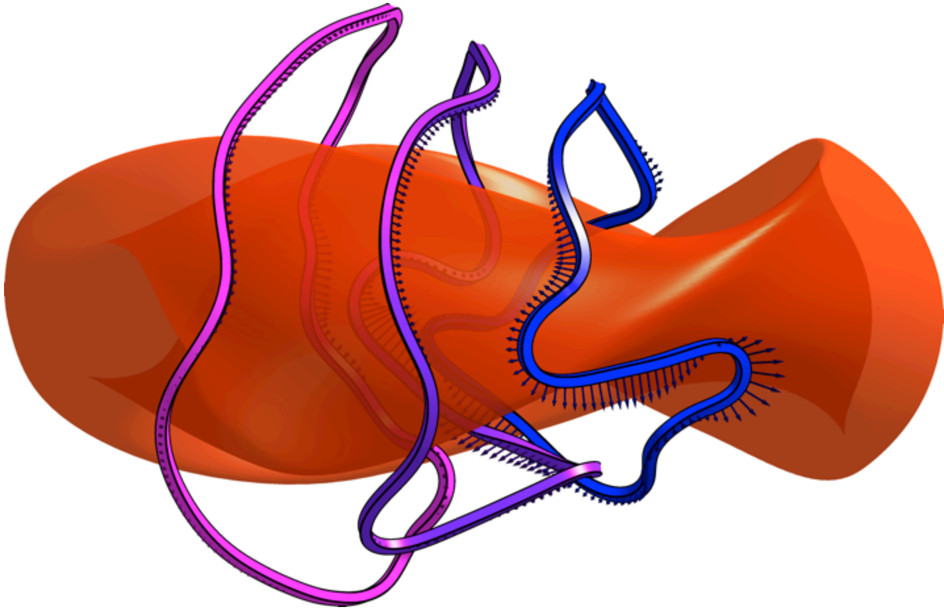




Example: Neoclassical transport $\varepsilon_{eff}^{3/2}$ at r/a=0.5

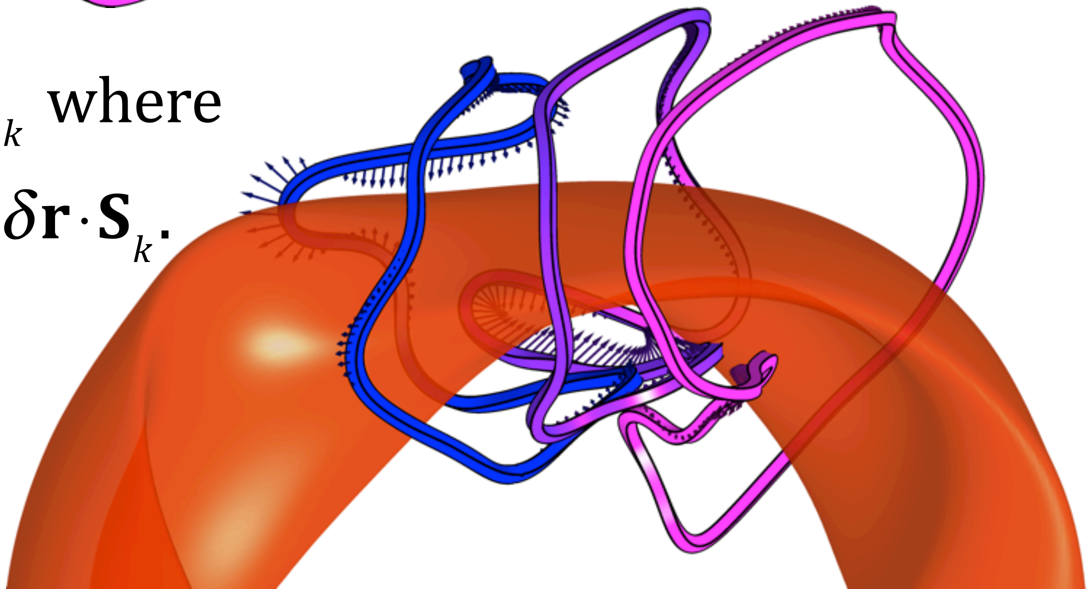
Shape gradient for coils:





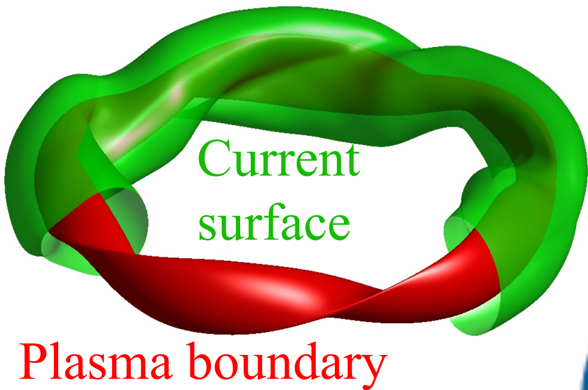
Arrows show $\mathbf{S}_k$ where

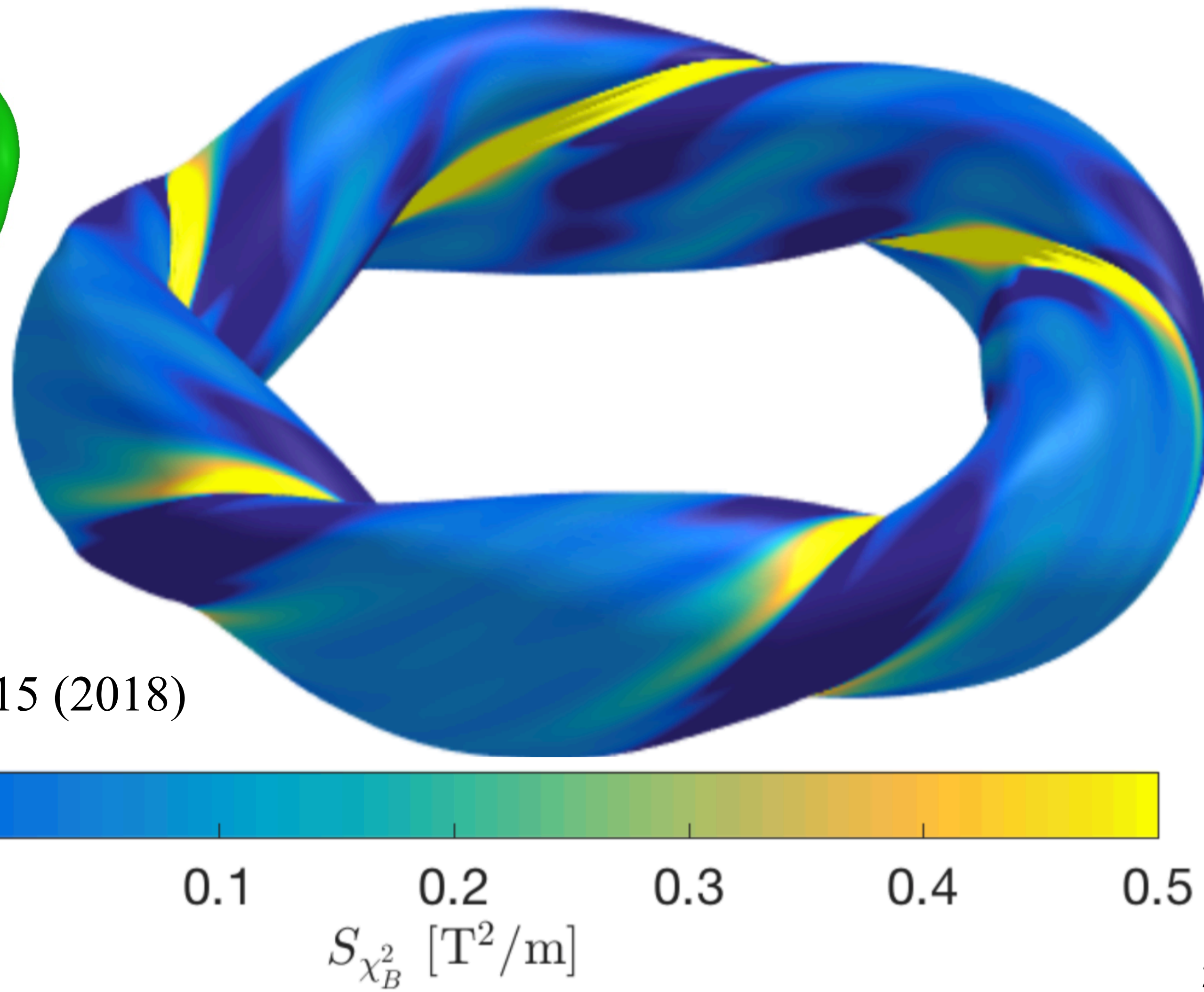
$$\delta f = \sum_{\text{coils } k} \int d\ell \; \delta\mathbf{r} \cdot \mathbf{S}_k .$$



The shape gradient can show where coils must be close to the plasma.

Shape gradient on a current surface for $f = \int d^2a \left(\mathbf{B}\cdot\mathbf{n}\right)^2$ given a fixed plasma boundary





E. J. Paul et al, *Nuclear Fusion* 58, 076015 (2018)

Understanding local sensitivity of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from parameter derivatives (e.g. STELLOPT)
- Examples
  - Rotational transform
  - Neoclassical transport
  - Coil-plasma distance
- Coil tolerances
- Magnetic sensitivity and tolerances

Coil tolerances can be computed from the shape gradient.

Choose an acceptable Δf & any weight $w(\ell) \geq 0$.

$$\text{Let } T(\ell) = \frac{w(\ell)\,\Delta f}{\sum \int d\ell'\, w(\ell') \left|\mathbf{S}(\ell')\right|}.$$

If $\left|\delta \mathbf{r}\right| \leq T$,

$$\left|\delta f\right| \;\leq\; \int d\ell \left|\mathbf{S}\cdot\delta\mathbf{r}\right| \;\leq\; \int d\ell \left|\mathbf{S}\right|\left|\delta\mathbf{r}\right| \;\leq\; \int d\ell \left|\mathbf{S}\right| T \;=\; \Delta f.$$

Coil tolerances can be computed from the shape gradient.

Choose an acceptable Δf & any weight $w(\ell) \geq 0$.

$$\text{Let } T(\ell) = \frac{w(\ell)\,\Delta f}{\sum \int d\ell'\, w(\ell') \left|\mathbf{S}(\ell')\right|}.$$

If $\left|\delta \mathbf{r}\right| \leq T$,

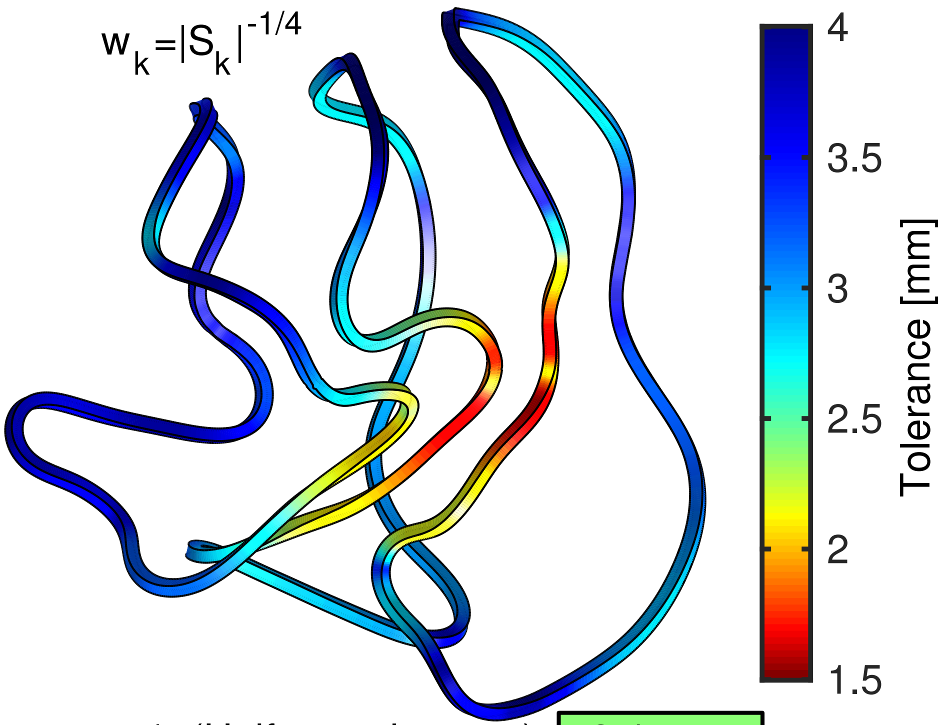
$$\left|\delta f\right| \leq \int d\ell \left|\mathbf{S}\cdot\delta\mathbf{r}\right| \leq \int d\ell \left|\mathbf{S}\right|\left|\delta\mathbf{r}\right| \leq \int d\ell \left|\mathbf{S}\right| T = \Delta f.$$

Conservative: a bound on the worst possible outcome.

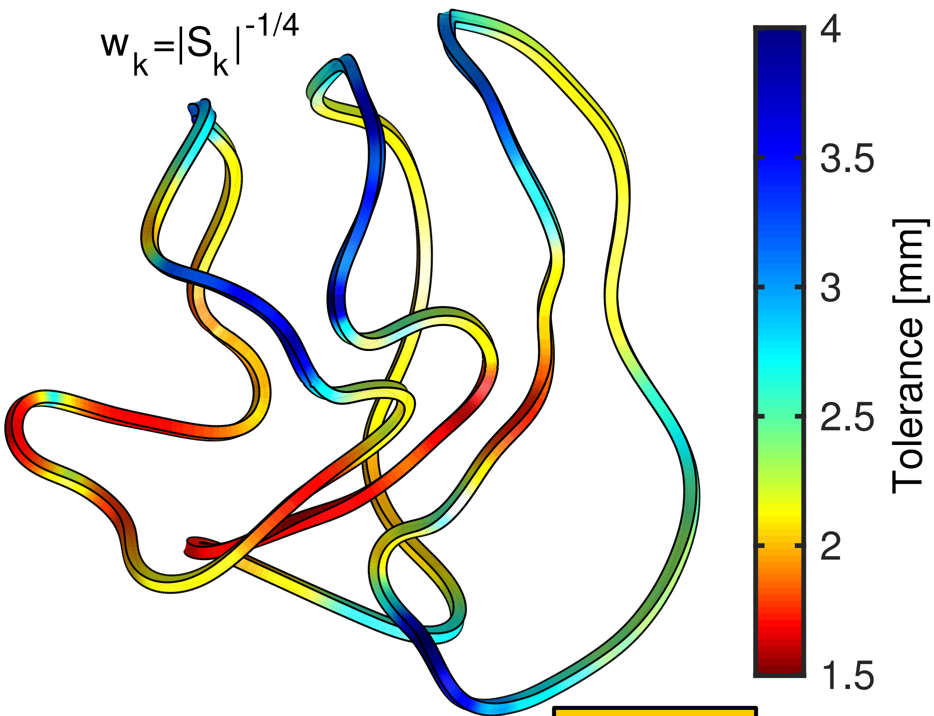
Coil tolerances can be computed from the shape gradient.

$$\Delta \iota = 0.02$$

$$\Delta \varepsilon_{eff}^{3/2} = \varepsilon_{eff}^{3/2} / 2$$



$w_k=1$ (Uniform tolerance): 2.4 mm



$w_k=1$ (Uniform tolerance): 2.0 mm

Understanding local sensitivity of stellarators using shape gradients

- 2 ways to represent derivatives: parameter derivatives vs. shape gradients.
- Computing shape gradients from parameter derivatives (e.g. STELLOPT)
- Examples
  - Rotational transform
  - Neoclassical transport
  - Coil-plasma distance
- Coil tolerances
- Magnetic sensitivity and tolerances

A magnetic sensitivity S_B can be computed from the shape gradient.

Define S_B by $\mathbf{B}_0 \cdot \nabla S_B = \langle S \rangle - S$.

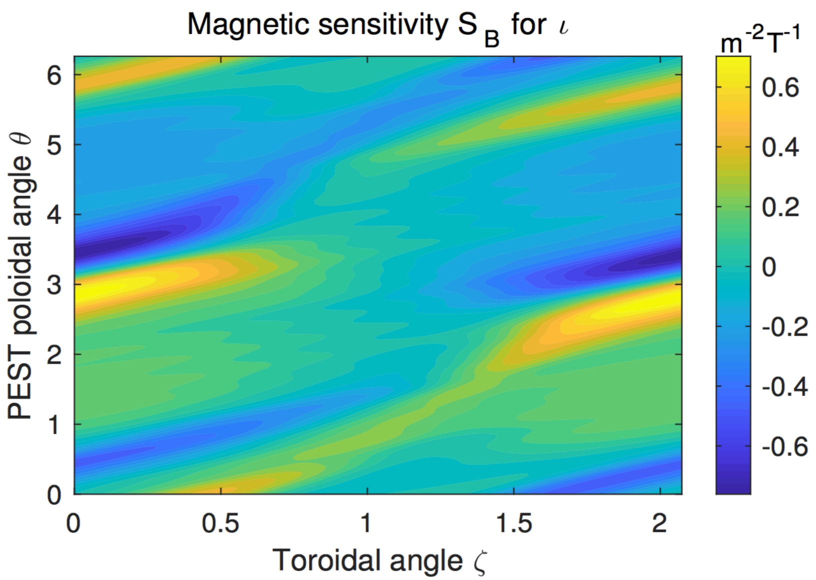
Substitute into $\delta f = \int d^2a\, S \delta\mathbf{r} \cdot \mathbf{n}$.

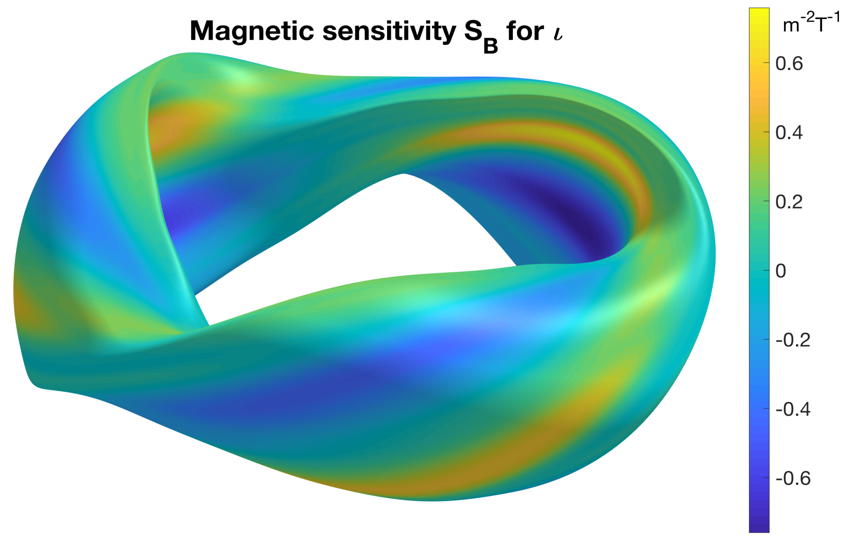
After some algebra ...

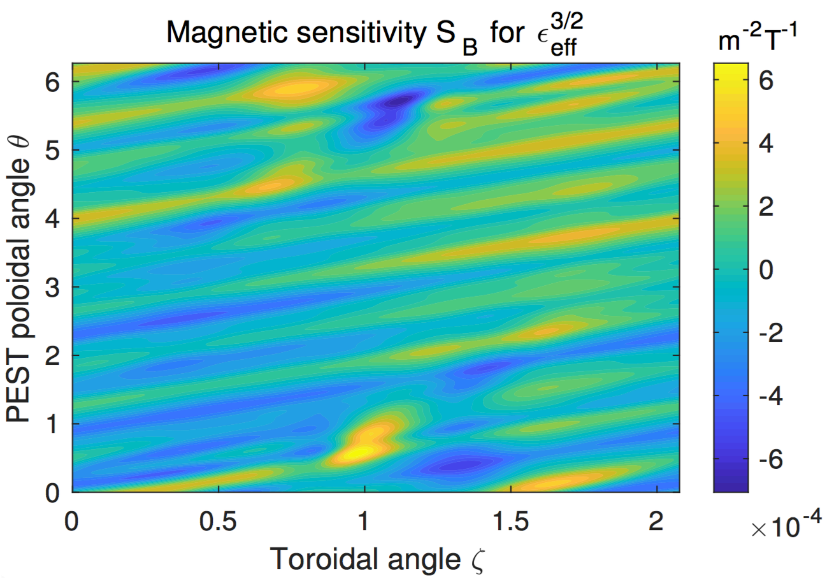
$$\Rightarrow \quad \delta f = \int d^2a\, S_B \delta\mathbf{B} \cdot \mathbf{n}.$$

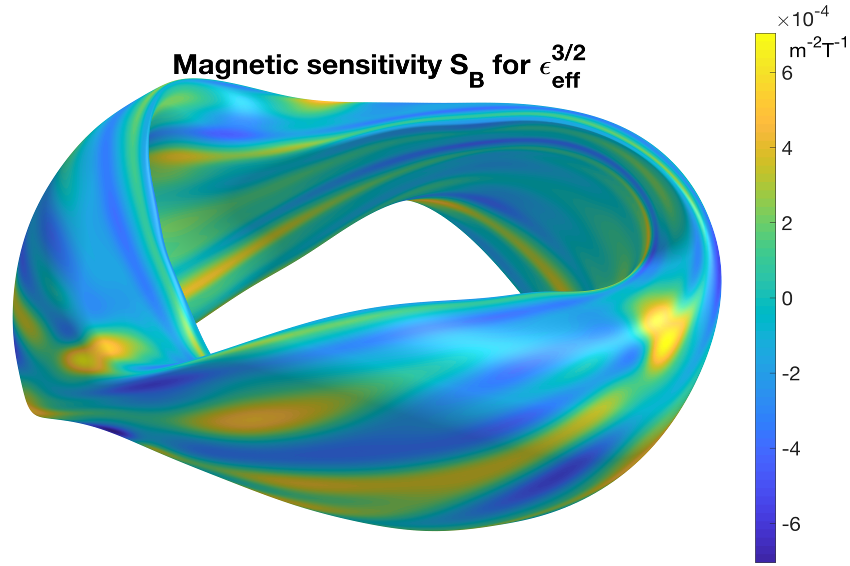
A magnetic sensitivity S_B can be computed from the shape gradient.

$$\delta f = \int d^2a\, S_B \delta \mathbf{B}\cdot\mathbf{n} \quad \text{where} \quad \mathbf{B}_0 \cdot \nabla S_B = \langle S \rangle - S.$$









A magnetic tolerance T_B can be computed from the magnetic sensitivity.

Choose an acceptable Δf & any weight $W(\theta,\zeta) \geq 0$.

$$\text{Let } T_B(\theta,\zeta) = \frac{W(\theta,\zeta)\,\Delta f}{\int d^2a'\, W(\theta',\zeta')\left|S_B(\theta',\zeta')\right|}.$$

If $\left|\delta\mathbf{B}\cdot\mathbf{n}\right| \leq T_B$,

$$\begin{aligned} \left|\delta f\right| &\leq \int d^2a \left|S_B\right|\left|\delta\mathbf{B}\cdot\mathbf{n}\right| \\ &\leq \int d^2a \left|S_B\right| T_B \\ &\leq \Delta f. \end{aligned}$$

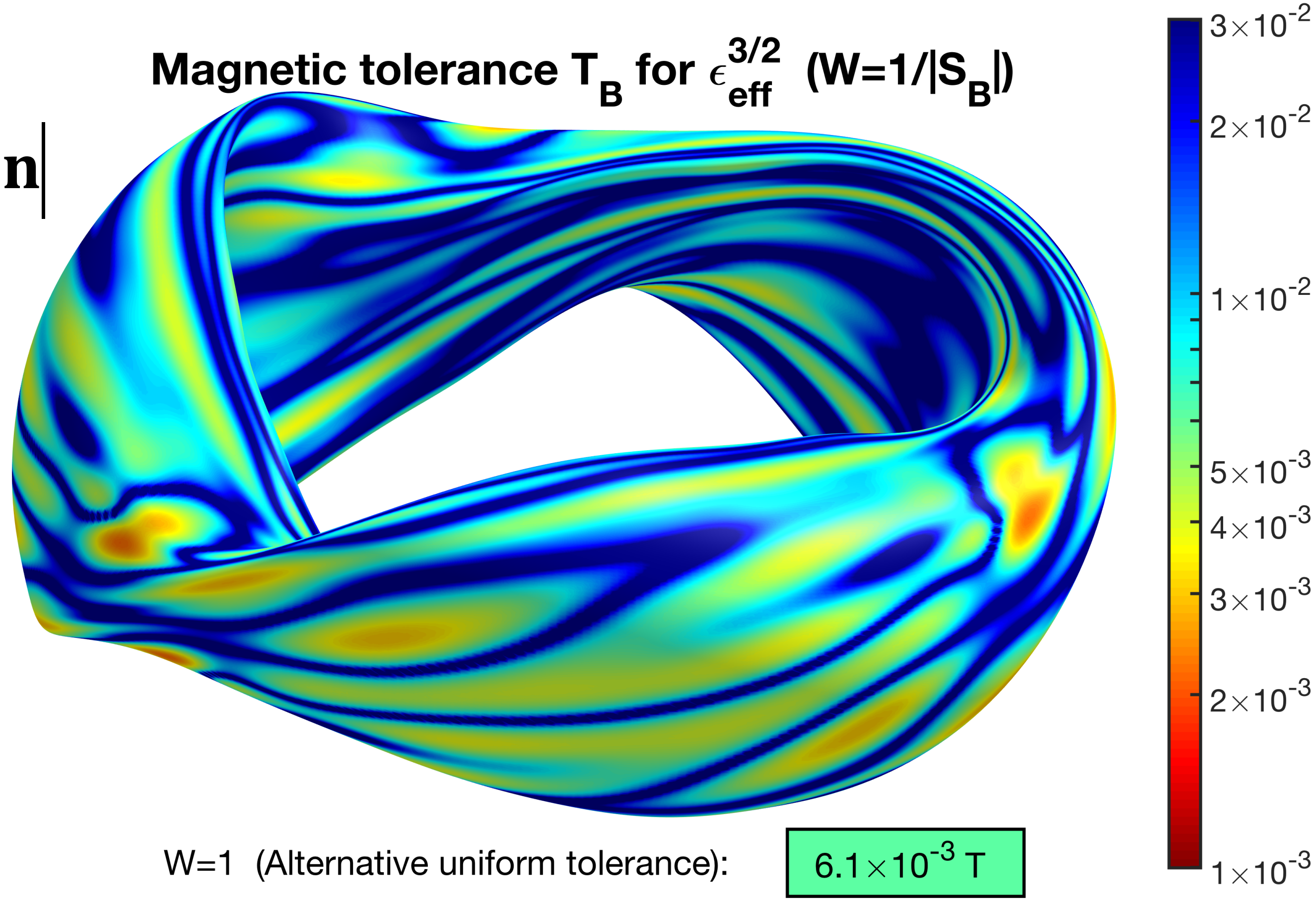
A magnetic tolerance T_B can be computed from the magnetic sensitivity.

Choose an acceptable Δf & any weight $W(\theta,\zeta)\geq 0$.

$$\text{Let } T_B(\theta,\zeta)=\frac{W(\theta,\zeta)\,\Delta f}{\int d^2a'\, W(\theta',\zeta')\left|S_B(\theta',\zeta')\right|}.$$

If $\left|\delta\mathbf{B}\cdot\mathbf{n}\right|\leq T_B$,

$$\begin{aligned}|\delta f| &\leq \int d^2a\,|S_B|\,|\delta\mathbf{B}\cdot\mathbf{n}| \\ &\leq \int d^2a\,|S_B|\,T_B \\ &\leq \Delta f.\end{aligned}$$



Conclusions

Shape gradients provide ***local*** sensitivity & tolerance information which can inform

- How accurately and rigidly the coils should be built,
- Where coils should be connected to support structure,
- Where sources of error fields like coil leads should be located,

and enable systematic calculation of tolerances.

Future work:

- Shape gradients for integrability/islands,
- Direct calculation of shape gradients using adjoint methods,
- Target tolerances in STELLOPT to increase them.

Nuclear Fusion ***58***, *076023 (2018)*

Extra slides

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Plasma boundary shape:

Parameters p_j are $\left\{R^c_{m,n}, Z^s_{m,n}\right\}$.

Compute $\dfrac{\partial f}{\partial p_j}$ using finite differences, e.g. STELLOPT.

Discretize shape gradient: $S(\theta,\zeta) = \sum_q S_q \cos(m_q\theta - n_q\zeta)$

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Plasma boundary shape:

Parameters p_j are $\left\{R^c_{m,n}, Z^s_{m,n}\right\}$.

Compute $\dfrac{\partial f}{\partial p_j}$ using finite differences, e.g. STELLOPT.

Discretize shape gradient: $S(\theta,\zeta) = \sum_q S_q \cos\left(m_q \theta - n_q \zeta\right)$

$$\int d^2a \left(\delta \mathbf{r} \cdot \mathbf{n}\right) S = \delta f \qquad \Rightarrow \qquad \text{Solve } \int d^2a \frac{\partial \mathbf{r}}{\partial p_j} \cdot \mathbf{n} S = \frac{\partial f}{\partial p_j} \text{ for } S. \qquad (1)$$

Linear system, not square.

The shape gradient can be computed from parameter derivatives by solving a small linear system.

Plasma boundary shape:

Parameters p_j are $\left\{R^c_{m,n}, Z^s_{m,n}\right\}$.

Compute $\dfrac{\partial f}{\partial p_j}$ using finite differences, e.g. STELLOPT.

Discretize shape gradient: $S(\theta,\zeta) = \sum_q S_q \cos\left(m_q\theta - n_q\zeta\right)$

$$\int d^2a \left(\delta\mathbf{r}\cdot\mathbf{n}\right) S = \delta f \qquad \Rightarrow \qquad \text{Solve } \int d^2a \frac{\partial \mathbf{r}}{\partial p_j}\cdot\mathbf{n} S = \frac{\partial f}{\partial p_j} \text{ for } S. \qquad (1)$$

Linear system, not square.

Check that $\dfrac{\partial f}{\partial p_j}$ is in the column space of matrix.

If so, (1) can be solved for S_q using pseudo-inverse of matrix.

Shape gradients have been used in other fields, mostly for shapes interacting with neutral fluid.

- O. Pironneau, *Optimal Shape Design for Elliptic Systems* (Springer-Verlag, 1984).
- K. K. Choi and N.-H. Kim, *Structural Sensitivity Analysis and Optimization, vol 1* (Springer, 2005).
- M. C. Delfour and J.-P. Zolesio, *Shapes and Geometries: Metrics, Analysis, Differential Calculus, and Optimization*, 2nd ed. (SIAM, 2011).
- …

And recently for optimizing tokamak divertor shapes:

- W. Dekeyser, Ph.D. thesis, KU Leuven (2014).
- W. Dekeyser et al, *Nucl. Fusion* 54, 073022 (2014).
- M. Baelmans, et al, *Nucl. Fusion* 57, 036022 (2017).

The shape gradient representation can be expected to exist for many important shape functionals.

Derivative of a function of *n* numbers $f\left(r_1, r_2, \ldots, r_n\right)$: $\quad \delta f = \sum_{j=1}^{n} \frac{\partial f}{\partial r_j} \delta r_j$

$$n \to \infty \text{ limit: } f = f\left[r\left(\vartheta\right)\right], \quad \delta f = \int_0^{2\pi} d\vartheta \frac{\delta f}{\delta r} \delta r$$

This is an instance of the *Riesz representation theorem*: any linear operator can be written as an inner product with some element of the appropriate space.

$$\text{3D: } f = f\left[r_X\left(\vartheta\right), r_Y\left(\vartheta\right), r_Z\left(\vartheta\right)\right], \quad \delta f = \sum_{i=X,Y,Z} \int_0^{2\pi} d\vartheta \frac{\delta f}{\delta r_i} \delta r_i$$

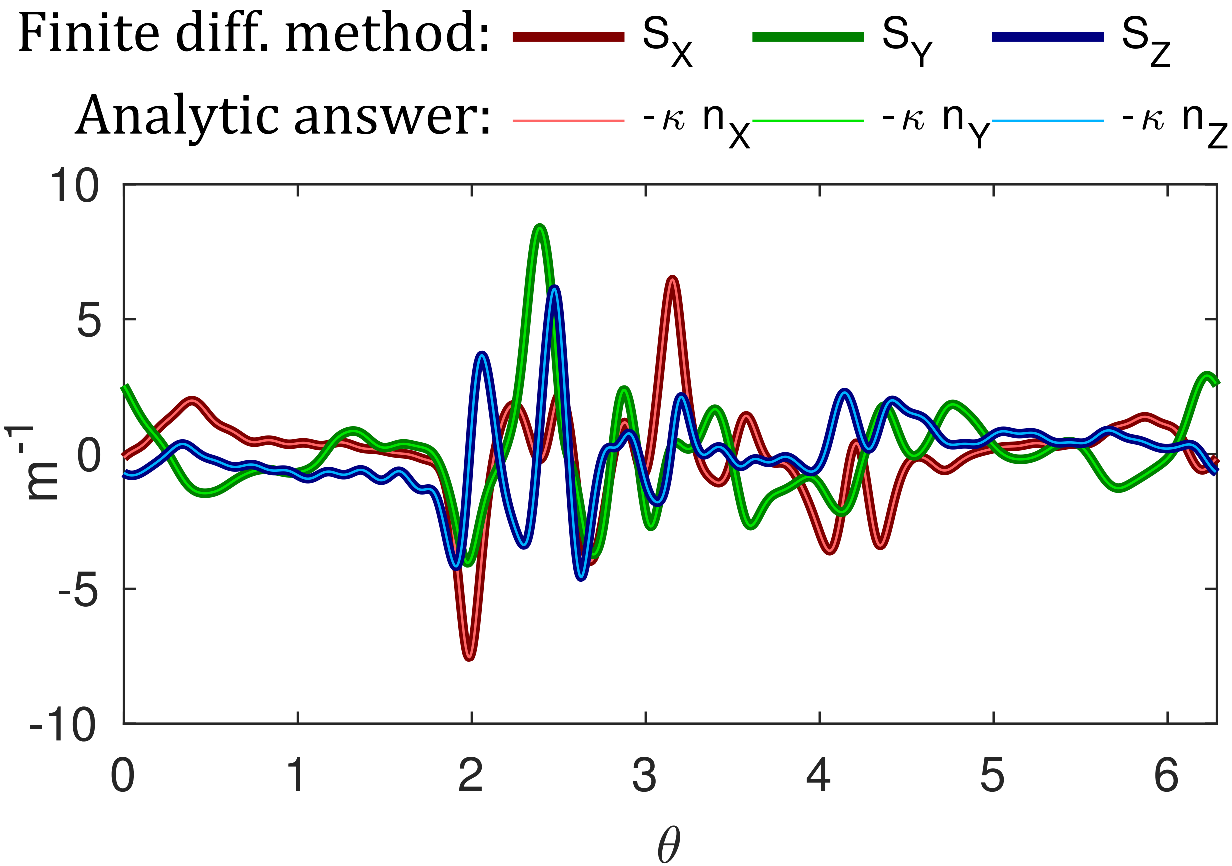
$$\text{Define } \mathbf{S} = \left|\frac{d\mathbf{r}}{d\vartheta}\right|^{-1} \left(\mathbf{e}_X \frac{\partial f}{\partial r_X} + \mathbf{e}_Y \frac{\partial f}{\partial r_Y} + \mathbf{e}_Z \frac{\partial f}{\partial r_Z} \right) \qquad \Rightarrow \quad \delta f = \int d\ell \, \mathbf{S} \cdot \delta \mathbf{r}$$

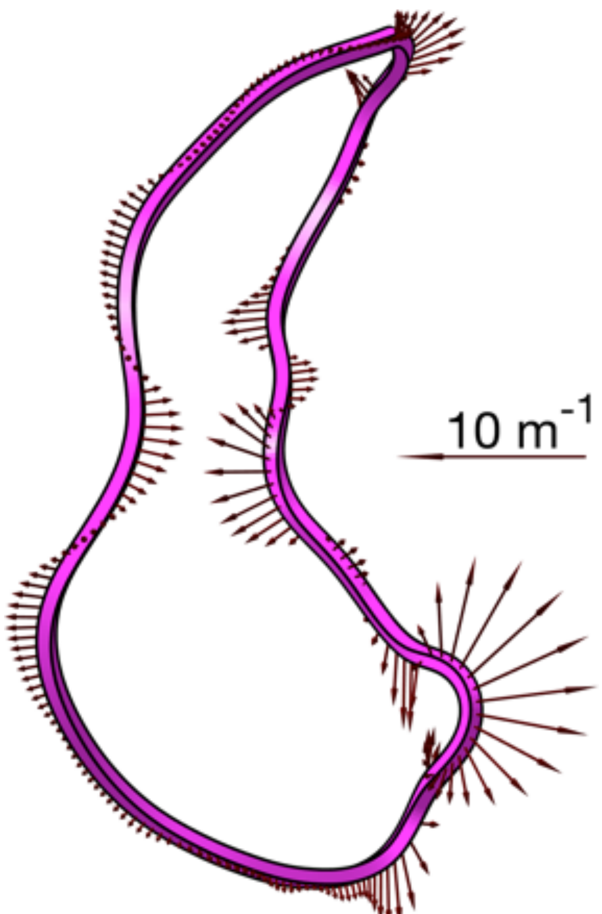
The algorithm for computing coil shape gradients can be verified by comparison to analytic theory.

Consider $f = \text{length}$. Analytic result: $\mathbf{S} = -\kappa \mathbf{n}$

(NCSX type-A coil)

Shape gradient **S**:





`Direct method' to compute sensitivity map

$$\int d^2a\, S_d\, \frac{\partial \mathbf{r}}{\partial p_k}\cdot\mathbf{n} = \frac{\partial f}{\partial p_k} \qquad p_k \in \left\{ R^c_{mn}, Z^s_{mn} \right\}$$

Unknowns = S_d on a grid: $S_d\left(\theta_j,\ \zeta_j\right)$

$$\underbrace{\left(\int d^2a\, \frac{\partial \mathbf{r}}{\partial p_k}\cdot\mathbf{n} \right)}_{\mathbf{D}} \underbrace{\left(S_d \right)}_{\mathbf{S}} = \underbrace{\left(\frac{\partial f}{\partial p_k} \right)}_{\mathbf{j}}$$

SVD: $\mathbf{D} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ Solution by pseudo-inverse: $\mathbf{S} = \mathbf{V}\mathbf{\Sigma}^+\mathbf{U}^T\mathbf{j}$

If $\mathbf{j}$ is not in the column space of $\mathbf{D}$, then no sensitivity map exists.

$\Rightarrow$ Check whether $\mathbf{U}^T\mathbf{j}$ entries are small where Σ entries are small.

Fourier method to compute sensitivity map

$$\int d^2a\, S_d \frac{\partial \mathbf{r}}{\partial p_k}\cdot \mathbf{n} = \frac{\partial f}{\partial p_k} \qquad p_k \in \left\{ R^c_{mn}, Z^s_{mn} \right\} \qquad S_d = \sum_{m,n} S_{m,n} \cos\left(m\theta - n\zeta \right)$$

Unknowns: $S_{m,n}$

$$\underbrace{\left(\int d^2a \left(\frac{\partial \mathbf{r}}{\partial p_k}\cdot \mathbf{n} \times \cos\left(m\theta - n\zeta \right) \right) \right)}_{\mathbf{D}} \underbrace{\left(S_{m,n} \right)}_{\mathbf{S}} = \underbrace{\left(\frac{\partial f}{\partial p_k} \right)}_{\mathbf{j}}$$

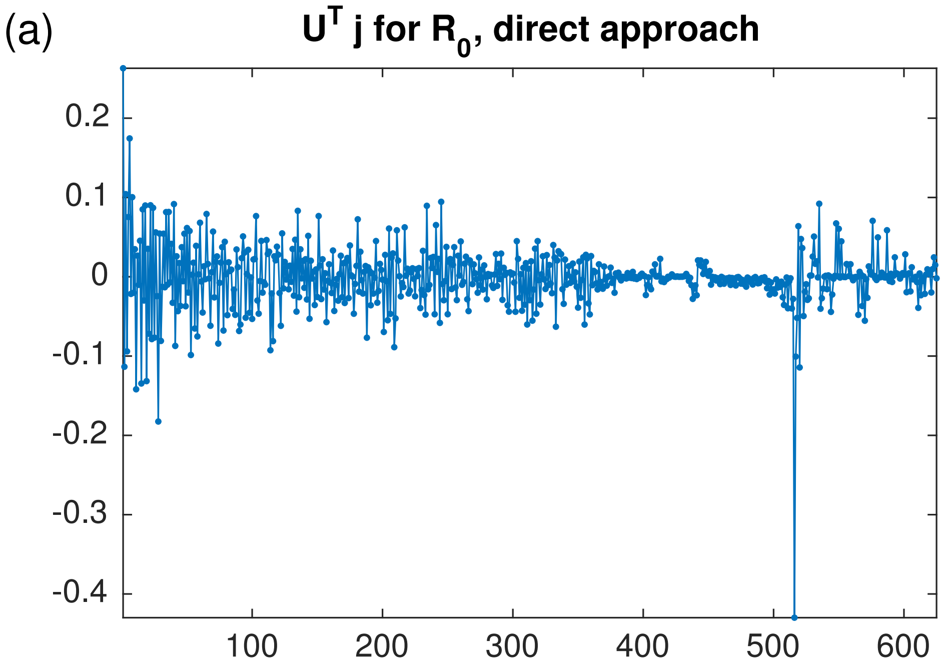
SVD: $\mathbf{D} = \mathbf{U}\Sigma\mathbf{V}^T$ Solution by pseudo-inverse: $\mathbf{S} = \mathbf{V}\Sigma^+\mathbf{U}^T\mathbf{j}$

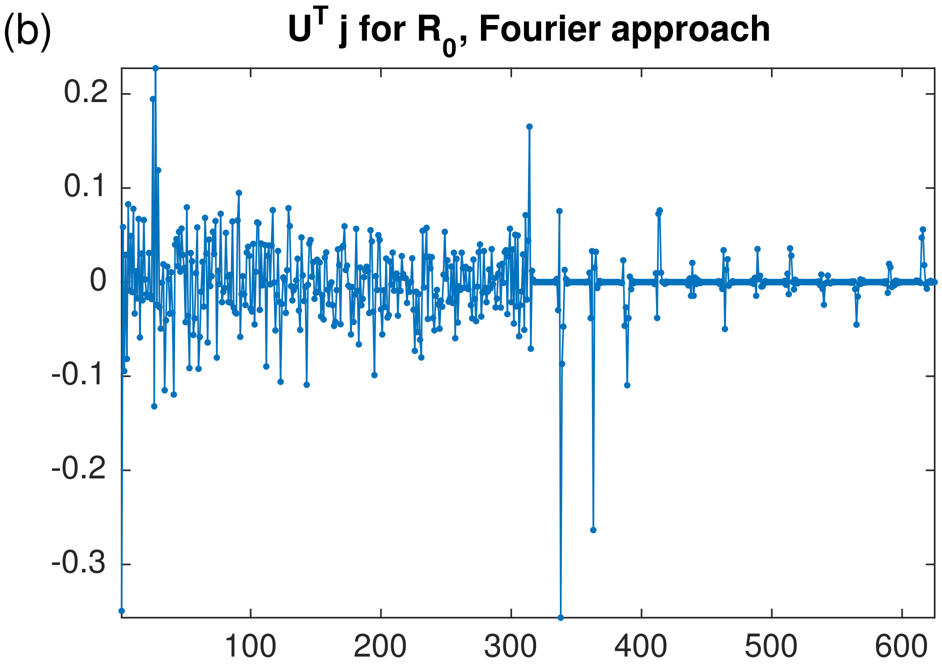
If $\mathbf{j}$ is not in the column space of $\mathbf{D}$, then no sensitivity map exists.

$\Rightarrow$ Check whether $\mathbf{U}^T\mathbf{j}$ entries are small where Σ entries are small.

If f is coordinate-dependent, so a sensitivity map does not exist, the RHS is not in the column space.

$$\text{Example: } f = R_0 = \frac{1}{(2\pi)^2}\int_0^{2\pi} d\theta \int_0^{2\pi} d\zeta\ R$$





A magnetic sensitivity map can be computed from the displacement sensitivity map.

$$\mathbf{B}=\mathbf{B}_0+\delta\mathbf{B},\quad \psi=\psi_0+\delta\psi \qquad \mathbf{B}\cdot\nabla\psi=0 \qquad \mathbf{B}_0\cdot\nabla\delta\psi+\delta\mathbf{B}\cdot\nabla\psi_0=0$$

$$0=d\psi=\delta\psi+\delta\mathbf{r}\cdot\nabla\psi_0 \qquad \Rightarrow \qquad \delta\psi=-\delta\mathbf{r}\cdot\nabla\psi_0$$

$$\boxed{\mathbf{B}_0\cdot\nabla\left(\delta\mathbf{r}\cdot\nabla\psi_0\right)=\delta\mathbf{B}\cdot\nabla\psi_0}$$

Equivalent to the $\nabla\psi_0$ component of the MHD induction equation $\delta\mathbf{B}=\nabla\times\left(\delta\xi\times\mathbf{B}_0\right)$.

Define S_B by $\boxed{\mathbf{B}_0\cdot\nabla S_B=\left\langle S_d\right\rangle-S_d.}$

$$\delta f=\int d^2a\, S_s\, \delta\mathbf{r}\cdot\mathbf{n} = \left\langle S_d\right\rangle\int d^2a\,\delta\mathbf{r}\cdot\mathbf{n}-\int d^2a\,\delta\mathbf{r}\cdot\mathbf{n}\mathbf{B}_0\cdot\nabla S_B$$

$$= \left\langle S_d\right\rangle\delta V+\int d^2a\,\frac{S_B}{\left|\nabla\psi_0\right|}\mathbf{B}_0\cdot\nabla\left(\delta\mathbf{r}\cdot\nabla\psi_0\right)$$

By parts

$$\boxed{\delta f = \left\langle S_d\right\rangle\delta V+\int d^2a\, S_B\delta\mathbf{B}\cdot\mathbf{n}}$$